

## CLAIMS

What is claimed is:

- 1 1. A method comprising:  
2       executing a speculative read-reordered load instruction;  
3       storing memory conflict information representing the speculative read-  
4 reordered load;  
5       matching an address of a potentially conflicting load against an address of  
6 the stored memory conflict information; and  
7       invalidating stored memory conflict information with a matching address.
- 1 2. The method of claim 1, wherein the stored memory conflict information is  
2 invalidated if the stored memory conflict information has a different value than  
3 the potentially conflicting load.
- 1 3. The method of claim 2, further comprising executing a read re-ordered  
2 load check instruction to determine the validity of the speculative read re-  
3 ordered load.
- 1 4. The method of claim 1, wherein the memory conflict information is stored  
2 in a read re-ordered load address table (RRLAT).
- 1 5. The method of claim 5, further comprising updating the stored memory  
2 conflict information by setting a validity bit in the RRLAT to a valid state when  
3 new memory conflict information is stored.

- 1 6. The method of claim 6, further comprising setting the validity bit to an  
2 invalid state if a later conflicting load operation is executed.
- 1 7. A processor, comprising:  
2 a RRLAT to store memory conflict information representing a speculative  
3 read re-ordered load; and  
4 a monitor to compare a potentially conflicting load against the stored  
5 memory conflict information.
- 1 8. The processor of claim 8, wherein the stored memory conflict information  
2 is invalidated if the stored memory conflict has a matching address.
- 1 9. The processor of claim 8, wherein the stored memory conflict information  
2 is invalidated if the stored memory conflict has a matching address and a  
3 different value than the potentially conflicting load.
- 1 10. The processor of claim 10, wherein the RRLAT is referenced upon the  
2 execution of a read re-ordered load check instruction to determine the validity of  
3 the speculative read re-ordered load.
- 1 11. The processor of claim 8, wherein the RRLAT may be any one of a  
2 direct-mapped, multi-way set associative, and fully associative data structure.
- 1 12. The processor of claim 8, wherein the RRLAT is portioned among  
2 hardware thread contexts.

1 13. The processor of claim 8, wherein the RRLAT includes storage  
2 locations for an address, a target register ID, a value, and validity information  
3 associated with the speculative read re-ordered load.

1 14. A computer system, comprising:  
2 a processor, including:  
3 a RRLAT to store memory conflict information representing a  
4 speculative read re-ordered load;  
5 a monitor to compare a potentially conflicting load against the  
6 stored memory conflict information, and to invalidate the stored memory conflict  
7 information if the stored memory conflict information has a matching address  
8 and a different value than the potentially conflicting load; and  
9 a cache memory.

1 15. The computer system of claim 16, wherein the monitor unit executes a  
2 read re-ordered load check instruction to determine the validity of the  
3 speculative read re-ordered load.

1 16. A computer system, comprising:  
2 a first processor; and  
3 a second processor, including:  
4 a RRLAT to store memory conflict information representing a  
5 speculative read re-ordered load received from the second processor; and  
6 a monitor to compare a potentially conflicting load received

7 from the first processor against the stored memory conflict information, and to  
8 invalidate the stored memory conflict information if the stored memory conflict  
9 information has a matching address and a different value than the potentially  
10 conflicting load.

1 17. The computer system of claim 19, wherein the monitor unit executes a  
2 read re-ordered load check instruction to determine the validity of the  
3 speculative read re-ordered load.

1 18. A computer system, comprising:  
2 a processor, including:  
3 a RRLAT to store memory conflict information representing a  
4 speculative read re-ordered load;  
5 a monitor to compare a potentially conflicting load against the  
6 stored memory conflict information, and to invalidate the stored memory conflict  
7 information if the stored memory conflict information has a matching address  
8 and a different value than the potentially conflicting load; and  
9 a cache memory; and  
10 a memory device coupled to the processor.

1 19. The computer system of claim 21, wherein the monitor unit validates  
2 stored memory conflict information with a matching address if the stored  
3 memory conflict information has a matching value to the potentially conflicting  
4 load.

1 20. The computer system of claim 22, wherein the monitor unit executes a  
2 read re-ordered load check instruction to test the validity of the speculative read  
3 re-ordered load.

1 21. The computer system of claim 21, further comprising a bus to control  
2 communications between the processor and the memory device.

1 22. A machine-readable medium storing a sequence of instructions that,  
2 when executed by a machine, cause the machine to:  
3 execute a speculative read-reordered load instruction;  
4 store memory conflict information representing the speculative read-  
5 reordered load;  
6 match the address of a potentially conflicting load against the address of  
7 the stored memory conflict information; and  
8 invalidate stored memory conflict information with a matching address if  
9 the stored memory conflict information has a different value than the potentially  
10 conflicting load.

1 23. The machine-readable medium of claim 25, the sequence of instructions,  
2 when executed by the computer system, further causing the computer system to  
3 validate stored memory conflict information with a matching address if the  
4 stored memory conflict information has a matching value to the potentially  
5 conflicting load.

1    24.    The machine-readable medium of claim 26, the sequence of instructions,  
2    when executed by the computer system, further causing the computer system to  
3    execute a read re-ordered load check instruction to determine the validity of the  
4    speculative read re-ordered load.